Install Visual Studio Code:

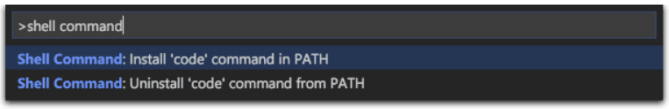
https://code.visualstudio.com

macOS

- Download the installer (Mac Universal, Stable Build).
- When the installer file has finished downloading, double click it to unzip it.
- Once it has unzipped, the file "Visual Studio Code.app" should appear in your Download folder. Drag and drop it into your Applications folder.

Additional step for macOS:

- Once installed, launch Visual Studio Code by double clicking on it from the Applications folder.
- Open the *Command Palette (Command+Shift+P)* and type **'shell command'** to find the **Shell Command: Install 'code' command in PATH** command, and select it.



Installing git, bash, command line tools, and homebrew:

About Git

- Git is what's known as a **version control system**.
- It allows software developers to store code in private and public repositories, track changes, collaborate with others, and more.
- You will learn more about using Git in the third course of this bootcamp, in the context of DevOps.
- Though we will not be using Git's version control features in this course, we will install and configure it now.

About Bash

- A shell is a software wrapper around an operating system. It provides a way to interface with the OS.
- Bash is a popular command-line-based shell that we will be using in this course. Bash is an acronym for Bourne-Again-Shell (an in-joke about its predecessor, the Bourne shell, named after its programmer).
- Originally created for GNU/Linux operating systems, bash can also be used in macOS and Windows.
- Bash comes pre-installed in macOS, though macOS users will need to configure their Terminal application to use it.
- Windows machines do not have bash built-in. When Windows users install Git, a tool called Git Bash will be installed along with it. This tool allows Windows users to use bash.

About Xcode Command Line Tools

- The Xcode Command Line Tools are a package from Apple that let your Mac use command line tools. You will need to install it to be able to use many development tools.

About Homebrew

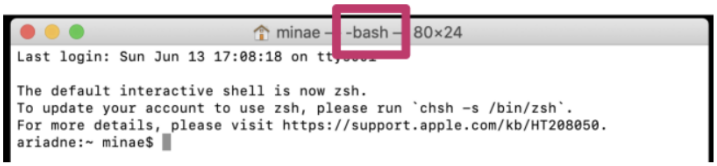
- Homebrew is a type of software called a package manager.
- Packages are bundled sets of files that are used by developers to create applications.
- Homebrew helps you download, install, and configure packages to work on your machine.

Update the default Terminal shell to bash

- Open the Terminal application.
  - The Terminal app comes installed by default on macOS. You can find it by opening your Launchpad and searching for "Terminal".
- When you first open Terminal, a new window will open.
- The current macOS default shell for Terminal is called **zsh**. This is a CLI that is very similar to bash. For the purposes of this course, we will have you switch to using **bash** in Terminal so that macOS and Windows students are using a consistent CLI.
- Switch your default terminal to **bash** by typing in the following command (then press enter):

```
chsh -s /bin/bash
```

- Enter your password if you are prompted for it.
- Close your Terminal window, then go to the Shell menu and select New Window.
- When the new window opens, you will see a message that says "The default interactive shell is now zsh." Don't panic! This is normal, albeit misleading.
- You should be able to see at the top of your Terminal that it says "bash", as shown below:



Install Xcode Command Line Tools

- Installing Homebrew first requires the Xcode Command Line Tools to be installed.
- Enter this command into your terminal:

```
xcode-select --install
```

- If your Mac already has this package installed, you will see an error message saying that it is already installed.
- If it is not already installed, you will be asked for your password. Enter it, then the tools will be installed.

Install Homebrew

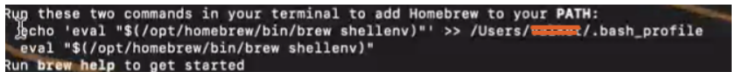
- Homebrew is a type of developer tool called a package manager, which helps with installing other tools.
- You will use the Terminal application to install Homebrew for macOS, which you will then use to install Git.
- To install Homebrew, copy the following two commands and paste them into your Terminal and press enter, one at a time:

```
touch ~/.bash_profile
```

```
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install.sh)"
```

- If you are asked for a password after entering the second of the two commands above, be aware that **you will most likely not be able to see anything being entered** into the terminal when you type in your password. This is for security reasons. Type in your password and press enter -- even if you can't see that you've typed anything. Press enter if directed to do so on the next screen.
- Homebrew should now be installed -- but if you are a M1 Mac or Linux user, READ ON!

- IMPORTANT!! ADDITIONAL STEP FOR M1 MAC AND LINUX USERS ONLY:
  - Mac users with the M1 (Apple Silicon) chip, and Linux users only! If that describes YOU, do not skip this! At the end of your installation, you should see a message that begins with "**Run these two commands in your terminal to add Homebrew to your PATH:**"
  - Below this, you will see two lines, one that begins with **echo** and one that begins with **eval**. The contents of these lines are customized to your specific system, but will look similar to this:



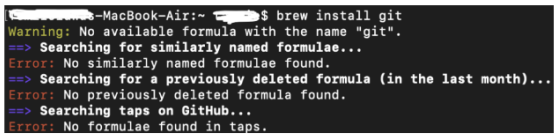
  - You *must* copy these two lines and paste them into your terminal, then press enter to run them. DO NOT MISS THIS PART!

Install Git

- Next, use Homebrew to install Git by entering this command into your Terminal:

```
brew install git
```

  - **TROUBLESHOOTING**: **If and only if** after you enter the above command, you see an error message like this:



    - Then the most likely issue is that the core tap for your homebrew installation has not been installed correctly. Enter the following two commands, one at a time, into your terminal:

```
rm -rf $(brew --repo homebrew/core)
```

```
brew tap homebrew/core
```

    - Then run the **brew install git** command once again. Make sure you do not see any error messages.
- Once the installation is complete, close your Terminal window and open a new one.

- You can check that your configuration is correct by typing the following at the prompt:

```
git config --global --list
```

  You should be able to see your user.name and user.email in the output.

Verify your Python installation

- Enter the following command into your terminal, **making sure to use a capital letter V**:

```
python -V
```

- The version shown should be 3.9.9.

Create github account:

https://github.com/

Download Github Desktop:

https://desktop.github.com/

Follow Me on Github and download Lessons_For_amy repository:

https://github.com/Jayden-58

Setup should now be complete!